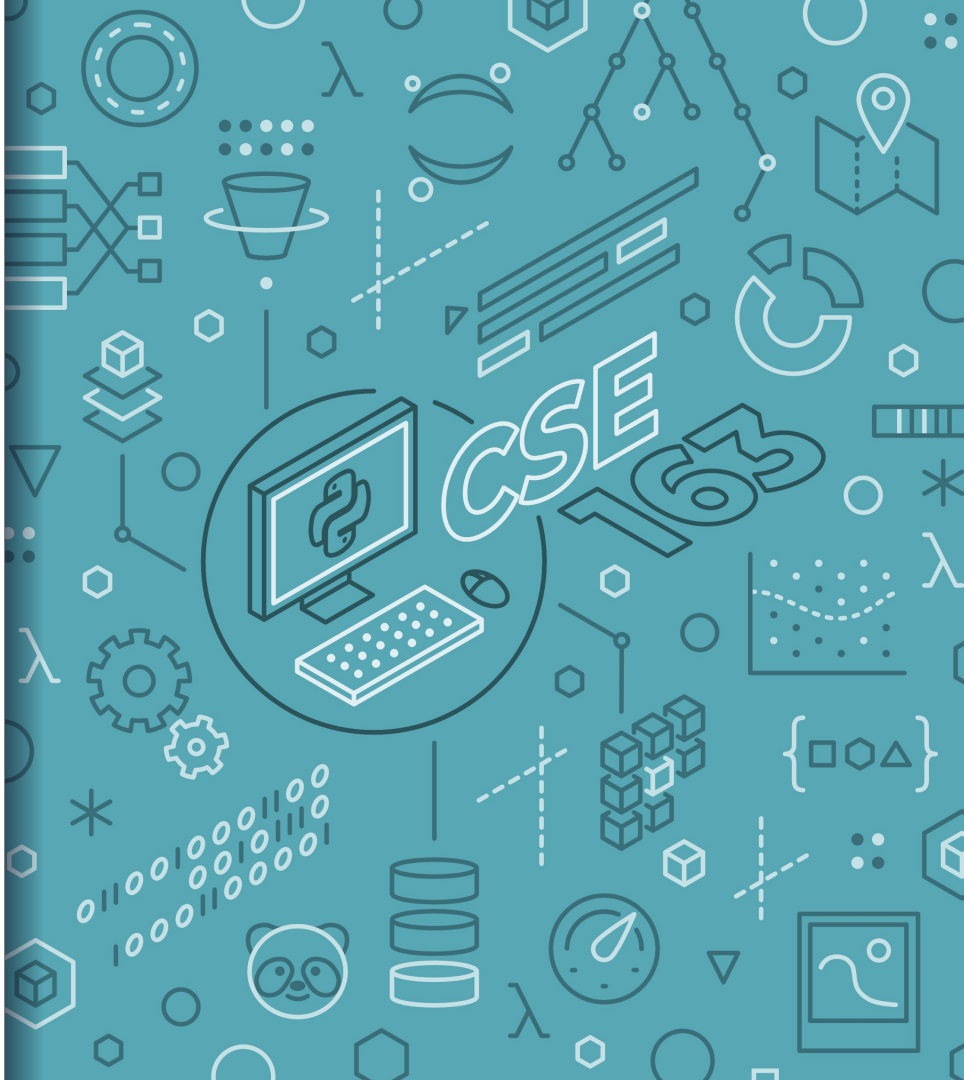
CSE 163

Machine Learning

Hunter Schafer



Cats vs Dogs

- Congrats! You have just been hired to work at Twitter as a data scientist to answer the age-old question:
  - Are cats or dogs the best?
- Given access to all the tweets, how might you tell?

```
cats = []
dogs = []
for tweet in tweets:
    if 'dog' in tweet:
        dogs.append(tweet)
    elif 'cat' in tweet:
        cats.append(tweet)
```

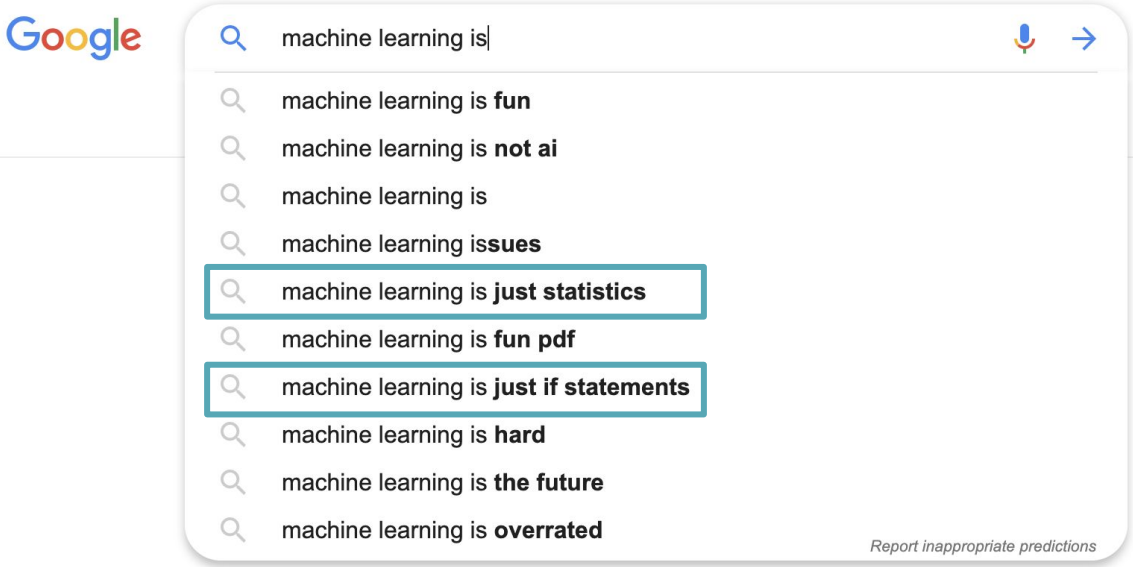
Cats vs Dogs

- The job market is good for data scientists specializing in cats-vs-dogs, so you leave Twitter to go work for Instagram
- Given access to all of the images, can we do the same?

```
cats = []
dogs = []
for pic in pictures:
    if dog in pic:
        dogs.append(pic)
    elif cat in pic:
        cats.append(pic)
```

Machine Learning

- What is Machine Learning?



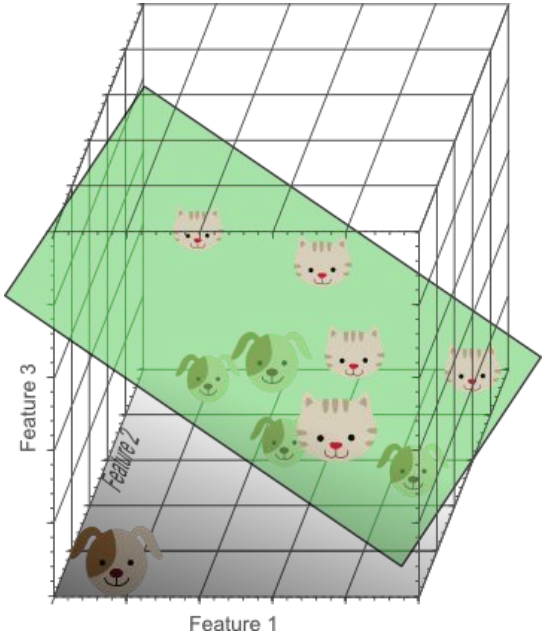
Machine Learning

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

Normal words: Uses data to automatically improve itself

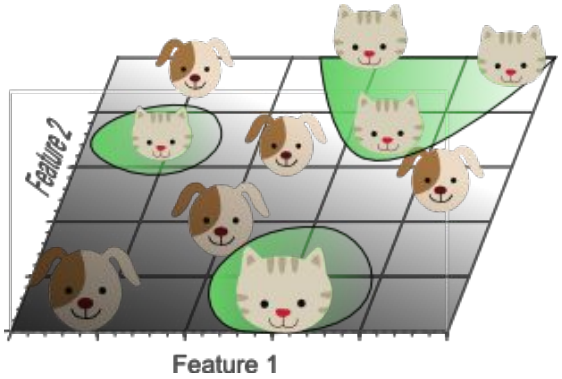
Learn Rules from Data

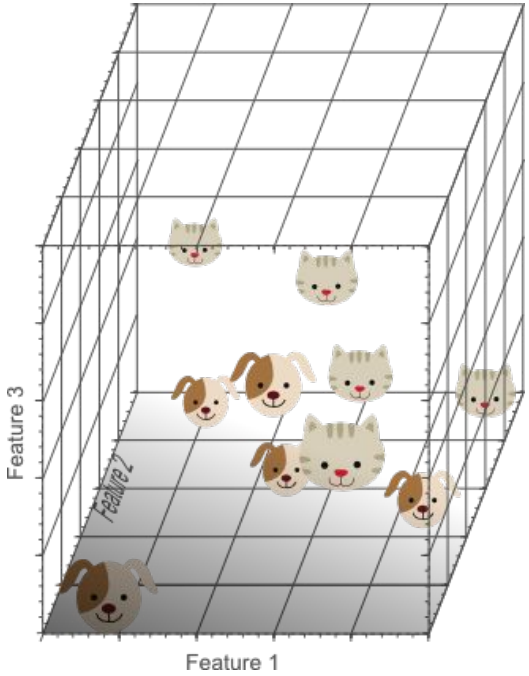
- Learn rules automatically from data, rather than programming
- We define **features** of our data that a model learns from to predict the **label**. We must learn this from a dataset, but the goal is to perform well on future, unseen data.



Features Matter

- There exists hundreds of different model types, but the most important thing is having good features to describe your data
  - Garbage In => Garbage Out





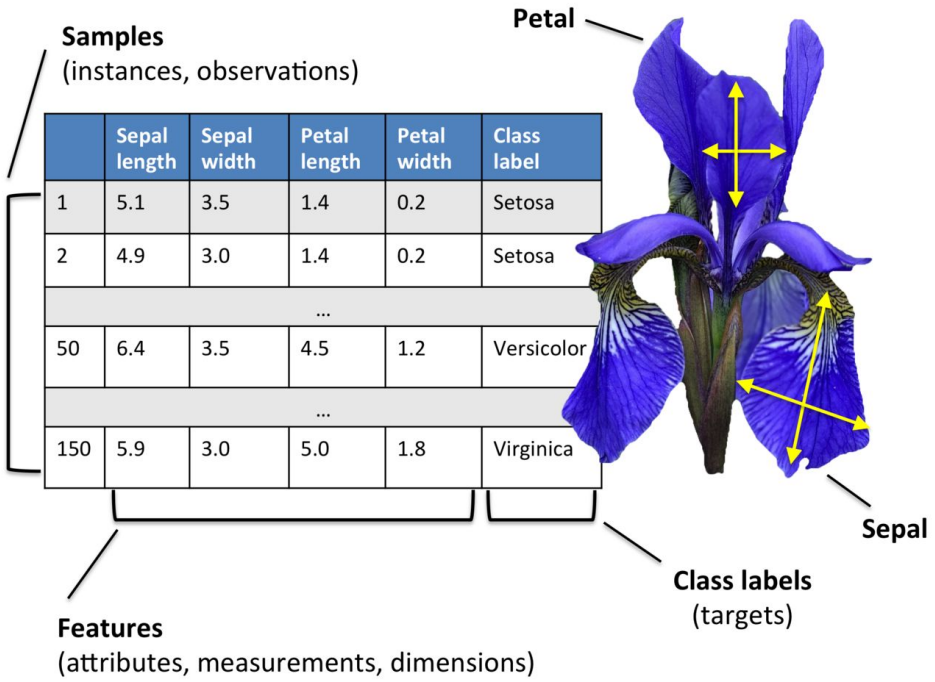
Categorizing ML

- **Supervised / Unsupervised**
  - Do you have to give the model labelled data to correct itself or can it do everything on its own?

- **Regression / Classification / Recommendation**
  - What you are trying to predict changes what models you use
  - Regression - Predicting a real number (price)
  - Classification - Predicting a category (cat or dog)
  - Recommendation - What movie should you watch next?

In lecture today, we will be focused on classification and your homework deals with regression

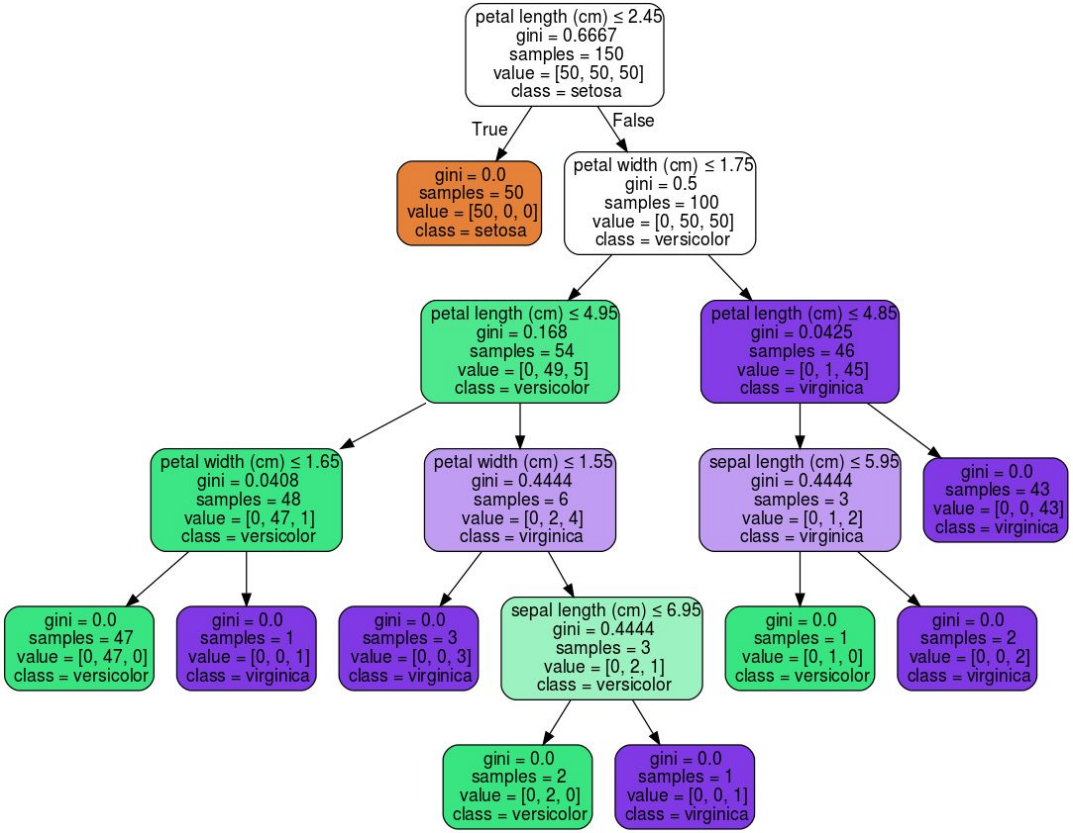
Iris Classification



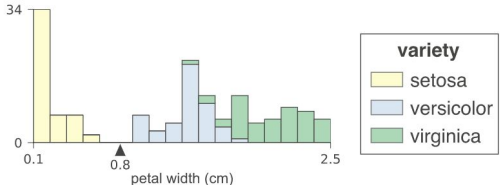
	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Demo

Decision Tree



Tree Splits


34
0
0.1
0.8
2.5
petal width (cm)
variety
setosa
versicolor
virginica

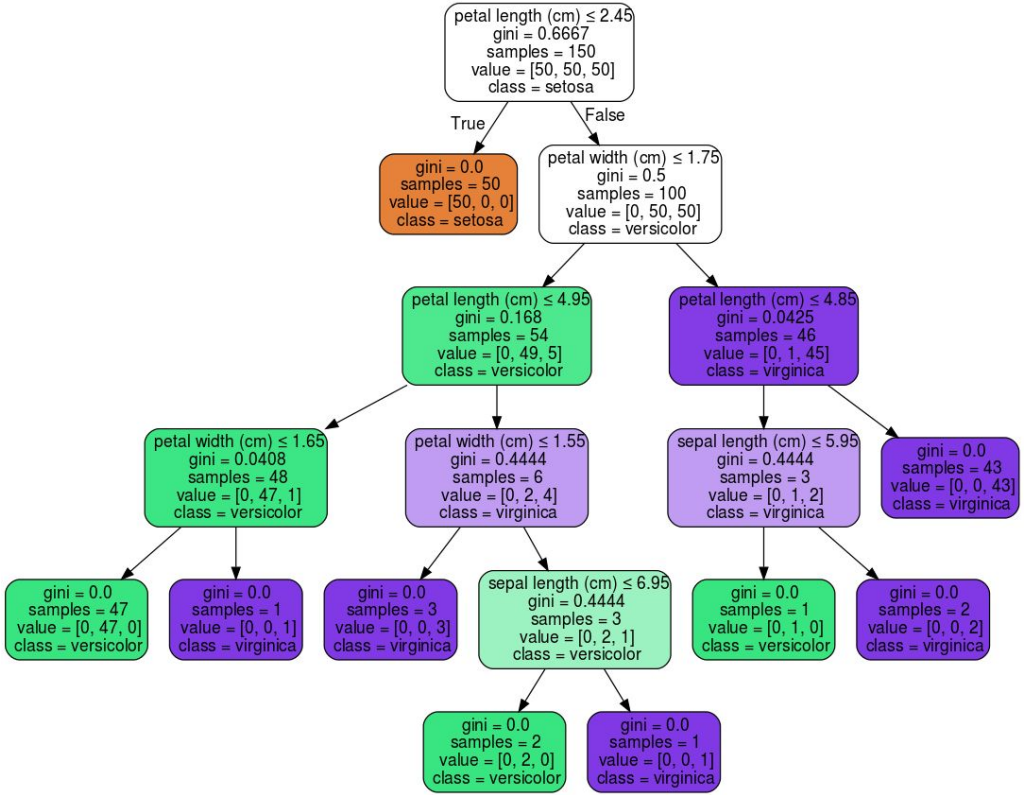
Poll Everywhere

Think

1 min

pollev.com/cse163

- What would you predict for a flower with
  - sepal length = 4 sepal width = 2
  - petal length = 5 petal width = 1.7

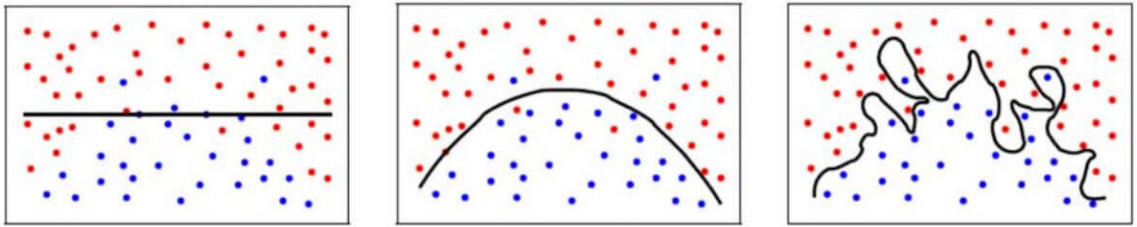


Titanic

Demo

Evaluating Models

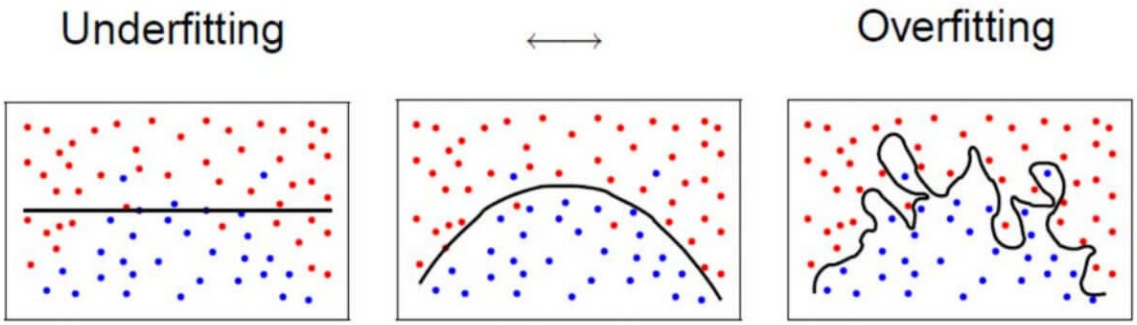
- Which model is best?



Reminder we want a model that will perform best on **future** data

Overfitting

- The most important problem in science you've never heard of
- Overfitting: When your model matches the training set so well, that it fails to generalize
  - Memorizing answers to Multiple Choice test
- Tall trees are likely to overfit if you don't have enough data
  - Can learn very complex boundaries
  - Very few points at the leaves



Evaluating Models

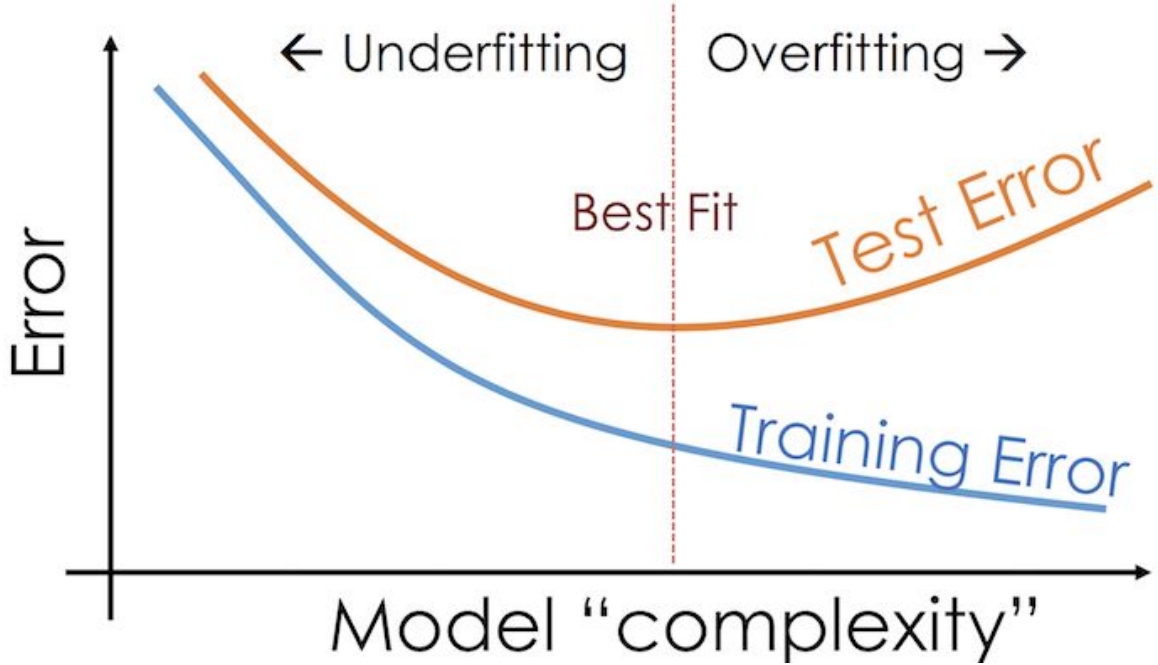
- Training is cool, but we want to know its future performance
- Training data can't be used to evaluate model
  - "I got 100% on the practice test I have been studying for 4 hours, therefore I will get 100% on the exam"
- Must hold out data called a **test set** to evaluate at the end
  - Unbiased estimate of performance in the wild

Never ever ever train or make decisions based on your test set.

If you do, it will no longer be good estimate of future performance.

Model Complexity

Note this is error, not accuracy (`error = 1 - accuracy`)



Recap

The important ideas

- Machine Learning
- Features and labels
  - String features have to be treated specially
- Models: Decision Tree
- Training
- Evaluating Model: Training accuracy vs test accuracy
- Overfitting
- Hyperparameter to change complexity (max height)

Scikit-learn (sklearn)

- `tree.DecisionTreeClassifier()`
  - `.fit(data, label)` and `.predict(data)`
- `metrics.accuracy_score(y_true, y_pred)`
- `model_selection.train_test_split(X, y, test_size)`